Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Second order interpolation of the flow field in the lattice Boltzmann method

J. Tölke *, M. Krafczyk

Institute for Computational Modeling in Civil Engineering, TU Braunschweig, Braunschweig, Germany

ARTICLE INFO

Keywords:
Lattice Boltzmann model
Interpolation

ABSTRACT

We propose a scheme to reconstruct the flow field up to second order in a cell composed of four lattices nodes in 2 dimensions. The information contained in the higher order moments of the distribution functions is used to construct an interpolation scheme of second order for the velocity field.

1. Introduction

The interpolation-supplemented lattice-Boltzmann equation is proved to recover the Navier–Stokes equation as long as a second order interpolation scheme for the distribution functions is used [1]. Using a linear interpolation scheme generates numerical viscosity degrading the lattice Boltzmann scheme. So lattice Boltzmann (LB) methods using grid refinement have to use interpolations of higher order to obtain suitable results [2–4]. This makes the algorithm more nonlocal, since for a higher order interpolation at least three coarse grid points have to be used and information from the next–next neighbor is needed. Here we propose a more local scheme avoiding this problem. The paper is organized as follows: In Section 2 we shortly sketch the LB method, in Section 3 we present a new interpolation scheme and in Section 4 the scheme is checked for consistency. Section 5 concludes the paper.

2. Lattice Boltzmann method

The Lattice Boltzmann method [5,6] is a numerical method to solve the Navier–Stokes equations, where mass fractions (also called particle distribution functions) propagate and collide on a regular grid. In the following $\mathbf{x}$ represents a two-dimensional vector in space and $\boldsymbol{f}$ a b-dimensional vector, where b is the number of microscopic velocities. We discuss the $d2q9$ model [7] with the following microscopic velocities,

$$\{\mathbf{e}_i, i = 0, \dots, 8\} = \left\{\begin{matrix} 0 & c & 0 & -c & 0 & c & -c & -c & c \\ 0 & 0 & c & 0 & -c & c & c & -c & -c \end{matrix}\right\} \tag{1}$$

generating a space-filling lattice with a nodal distance $h = c\Delta t$, where c is a constant microscopic velocity and Δt the time step. The lattice Boltzmann equation is

$$f_i(t + \Delta t, \mathbf{x} + \mathbf{e}_i \Delta t) = f_i(t, \mathbf{x}) + \Omega_i, \quad i = 0, \dots, 8 \tag{2}$$

where f_i are mass fractions (with unit kg m^{-3}) propagating with speed $\mathbf{e}_i$ and Ω is the collision operator. We use a multi-relaxation time MRT model [8–10] with a modified transformation matrix yielding simpler expressions for the collision operator [11]. The collision operator is

$$\Omega = \mathsf{M}^{-1}\boldsymbol{k}, \tag{3}$$

where M is the transformation matrix given in Appendix and $\boldsymbol{k}$ is the change of mass fractions in moment space.

* Corresponding author.
E-mail address: toelke@irmb.tu-bs.de (J. Tölke).

0898-1221/$ – see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2009.02.012

The moments $\boldsymbol{m}$ of the mass fractions are given with

$$\boldsymbol{m} = \mathsf{M}\boldsymbol{f} := (\rho, \rho_0 u_x, \rho_0 u_y, e, p_{xx}, p_{xy}, h_x, h_y, \kappa), \tag{4}$$

where ρ is a density variation, ρ_0 is a constant reference density and $(\rho_0 u_x, \rho_0 u_y)$ is the momentum. The moments e, p_{xx}, p_{xy} of second order, h_x, h_y of third order and κ of fourth order are related to the flow field by Eq. (7). Vector $\boldsymbol{k}$ is given with

$$\begin{aligned}
&k_0 = 0, \qquad k_1 = 0, \qquad k_2 = 0\\
&k_3 = k_e = -s_e\left(e - 3\,\rho_0(u_x^2 + u_y^2)\right)\\
&k_4 = k_{xx} = -s_\nu\left(p_{xx} - \rho_0(u_x^2 - u_y^2)\right)\\
&k_5 = k_{xy} = -s_\nu\left(p_{xy} - \rho_0\, u_x\, u_y\right)\\
&k_6 = k_{hx} = -s_h\, h_x\\
&k_7 = k_{hy} = -s_h\, h_y\\
&k_8 = k_\kappa = -s_\kappa \kappa,
\end{aligned} \tag{5}$$

where s_ν is a relaxation rate related to the viscosity and s_e, s_h and s_κ are relaxation rates related to the higher order moments.

Performing either a Chapman–Enskog [12,5] or an asymptotic expansion [13,14] of Eq. (2), where the time step Δt (and therefore the mesh spacing) as well as the Mach number go to zero, it can be shown that the LB-Method is a scheme of first order in time and second order in space for the incompressible Navier–Stokes equations. To couple the mesh spacing to the Mach number (diffusive scaling), the microscopic velocity c is defined as

$$c = \frac{c_0\, h_0}{h} \tag{6}$$

where h_0 and c_0 are a constant reference length and velocity. In the lattice Boltzmann context they are usually both set to one. The time step is related to these values by $\Delta t = h/c = h^2/(c_0\, h_0)$.

The analysis yields solutions of the moments $e, p_{xx}, p_{xy}, h_x, h_y$ and κ up to second order in terms of density and momentum [11] (for an expansion in distribution space using the advective scaling see [15]):

$$\begin{aligned}
&e = 3\,\rho_0(u_x^2 + u_y^2) + \mathcal{O}\left(h^2\right)\\
&p_{xy} = \rho_0\, u_x\, u_y - \frac{c^2\,\Delta t\,\rho_0}{3 s_\nu}(\partial_x\, u_y + \partial_y\, u_x) + \mathcal{O}\left(h^2\right)\\
&p_{xx} = \rho_0\,(u_x^2 - u_y^2) - \frac{2c^2\,\Delta t\,\rho_0}{3 s_\nu}(\partial_x\, u_x - \partial_y\, u_y) + \mathcal{O}\left(h^2\right)\\
&h_x = \frac{c^2\,\Delta t\,\rho_0}{s_h}(-\partial_x\, u_y^2 - 2\,\partial_y\, u_x u_y + 2\,\nu\,\partial_{yy}\, u_x - 4\,\nu \partial_{xx}\, u_x) + \mathcal{O}\left(h^2\right)\\
&h_y = \frac{c^2\,\Delta t\,\rho_0}{s_h}(-2\,\partial_x\, u_x u_y - \partial_y\, u_x^2 + 2\,\nu \partial_{xx}\, u_y - 4\,\nu \partial_{yy}\, u_y) + \mathcal{O}\left(h^2\right)\\
&\kappa = 0 + \mathcal{O}\left(h^2\right).
\end{aligned} \tag{7}$$

The kinematic viscosity is related to the relaxation rate s_ν by

$$\nu = \frac{1}{3}\left(\frac{1}{s_\nu} - \frac{1}{2}\right) c^2\,\Delta t. \tag{8}$$

The hydrodynamic pressure is given by

$$p = \frac{c^2}{3}\,\rho. \tag{9}$$

Collision rates s_e, s_h and s_κ are not relevant for the incompressible limit of the Navier–Stokes equations and can be chosen in the range $[0, 2]$. They can be tuned to improve stability [9], where the optimal values depend on the specific system under consideration (geometry, initial and boundary conditions) and can not be computed in advance. For high Reynolds-number flow a good choice is to set these values to one, for low Reynolds-number flow the magic combination given in is suitable [15]. A good choice is to set these values to one.

We can solve Eq. (7) for the elements $\epsilon_{\alpha\beta}$ of the strain rate tensor:

$$\begin{aligned}
&\partial_x\, u_x = \epsilon_{xx} = \frac{s_\nu}{c^2\,\Delta t}\left(\frac{3}{4}\left(u_x^2 - u_y^2 - \frac{p_{xx}}{\rho_0}\right)\right) + \mathcal{O}\left(h^2\right)\\
&\partial_y\, u_y = \epsilon_{yy} = \frac{s_\nu}{c^2\,\Delta t}\left(\frac{3}{4}\left(-u_x^2 + u_y^2 + \frac{p_{xx}}{\rho_0}\right)\right) + \mathcal{O}\left(h^2\right)\\
&\partial_x\, u_y + \partial_y\, u_x = 2\epsilon_{xy} = 3\,\frac{s_\nu}{c^2\,\Delta t}\left(u_x\, u_y - \frac{p_{xy}}{\rho_0}\right) + \mathcal{O}\left(h^2\right).
\end{aligned} \tag{10}$$

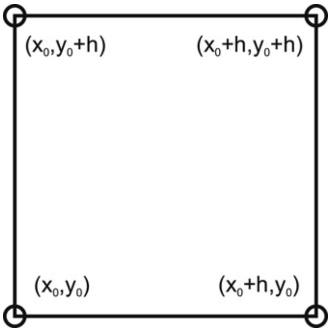

Fig. 1. Cell composed of four lattice nodes.

The velocity gradient matrix is not a tensor and is composed of the strain rate tensor $\epsilon_{\alpha\beta}$ and the rotation tensor $r_{\alpha\beta} = \partial_\beta\, u_\alpha - \partial_\alpha\, u_\beta$:

$$\partial_\beta\, u_\alpha = \epsilon_{\alpha\beta} + \frac{1}{2} r_{\alpha\beta}. \tag{11}$$

The diagonal entries of $r_{\alpha\beta}$ are zero and in two space dimensions the off-diagonal element is the vorticity:

$$\begin{pmatrix} \partial_x\, u_x & \partial_y\, u_x \\ \partial_x\, u_y & \partial_y\, u_y \end{pmatrix} = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xy} \\ \epsilon_{xy} & \epsilon_{yy} \end{pmatrix} + \frac{1}{2} \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix}. \tag{12}$$

With Eq. (7) it is not possible to compute the elements $\partial_y\, u_x$, $\partial_x\, u_y$ of the velocity gradient matrix separately or the vorticity $\omega = \partial_x\, u_y - \partial_y\, u_x$ with the local information contained in the moments. So it is not possible to do a higher order interpolation of the velocity between two nodes, since information is missing. In Section 3 we propose a non-local method using a cell composed of four nodes to obtain these entries.

3. Second order interpolation of the flow field

We consider a cell composed of four lattice nodes as shown in Fig. 1. The following ansatz for the velocity field $\mathbf{u}_a$ in the cell is chosen

$$\begin{aligned} u_{a,x}(x', y') &= a_0 + a_1\, x' + a_2\, y' + a_3\, x'\, y' + c_x(1 - x'^2) + c_y(1 - y'^2) \\ u_{a,y}(x', y') &= b_0 + b_1\, x' + b_2\, y' + b_3\, x'\, y' + d_x(1 - x'^2) + d_y(1 - y'^2), \end{aligned} \tag{13}$$

where

$$x' = \frac{2(x - x_0)}{h} - 1, \qquad y' = \frac{2(y - y_0)}{h} - 1, \tag{14}$$

and with a first set of 8 unknown coefficients $\{a_i, b_i | i = 0, 1, 2, 3\}$ and a second set of 4 unknown coefficients $\mathbf{g} = (c_i, d_i | i = 0, 1)$ This ansatz is of second order and the first set of coefficients is related to a simple bilinear interpolation and the second set introduces the quadratic terms. The special form of this ansatz guarantees that the velocity field is matched exactly at the 4 nodes.

The first set of coefficients is determined by demanding at each node of the cell

$$\begin{aligned} u_{a,x}(x_m, y_m) &= u_x(x_m, y_m) \\ u_{a,y}(x_m, y_m) &= u_y(x_m, y_m), \quad m = 1, \ldots, 4 \end{aligned} \tag{15}$$

and results in a simple bilinear interpolation. In the following the values x_0 and y_0 are set to zero to simplify the notation. The first set of coefficients $\{g_{\alpha,i},\, i = 0, \ldots, 3,\ \alpha = x, y\}$ with $g_x = a$ and $g_y = b$ is then given by

$$\begin{pmatrix} g_{\alpha,0} \\ g_{\alpha,1} \\ g_{\alpha,2} \\ g_{\alpha,3} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} u_\alpha(0,0) \\ u_\alpha(h,0) \\ u_\alpha(h,h) \\ u_\alpha(0,h) \end{pmatrix}, \quad \alpha = x, y. \tag{16}$$

Now we can proceed to determine the second set of coefficients. The elements of the strain rate tensor can be computed locally at the four nodes using Eq. (10). At each node of the cell we require now

$$\begin{aligned} \partial_x\, u_{a,x}(x_m, y_m) &= \epsilon_{xx}(x_m, y_m) \\ \partial_y\, u_{a,y}(x_m, y_m) &= \epsilon_{yy}(x_m, y_m) \\ \partial_y\, u_{a,x}(x_m, y_m) + \partial_x\, u_{a,y}(x_m, y_m) &= 2\epsilon_{xy}(x_m, y_m), \quad m = 1, \ldots, 4. \end{aligned} \tag{17}$$

This yields an equation system $\mathsf{C}\boldsymbol{g} = \boldsymbol{r}$ of 12 equations

$$\begin{pmatrix} \frac{4}{h} & 0 & 0 & 0 \\ 0 & 0 & \frac{4}{h} & 0 \\ 0 & \frac{4}{h} & 0 & \frac{4}{h} \\ -\frac{4}{h} & 0 & 0 & 0 \\ 0 & 0 & \frac{4}{h} & 0 \\ 0 & \frac{4}{h} & 0 & -\frac{4}{h} \\ -\frac{4}{h} & 0 & 0 & 0 \\ 0 & 0 & -\frac{4}{h} & 0 \\ 0 & -\frac{4}{h} & 0 & -\frac{4}{h} \\ \frac{4}{h} & 0 & 0 & 0 \\ 0 & 0 & -\frac{4}{h} & 0 \\ 0 & -\frac{4}{h} & 0 & \frac{4}{h} \end{pmatrix} \begin{pmatrix} c_x \\ c_y \\ d_y \\ d_x \end{pmatrix} = \begin{pmatrix} \frac{u_x(0,0) - u_x(h,0)}{h} + \epsilon_{xx}(0,0) \\ \frac{u_y(0,0) - u_y(0,h)}{h} + \epsilon_{yy}(0,0) \\ \frac{u_x(0,0) - u_x(0,h) + u_y(0,0) - u_y(h,0)}{h} + 2\,\epsilon_{xy}(0,0) \\ \frac{u_x(0,0) - u_x(h,0)}{h} + \epsilon_{xx}(h,0) \\ \frac{u_y(h,0) - u_y(h,h)}{h} + \epsilon_{yy}(h,0) \\ \frac{u_x(h,0) - u_x(h,h) + u_y(0,0) - u_y(h,0)}{h} + 2\,\epsilon_{xy}(h,0) \\ \frac{u_x(0,h) - u_x(h,h)}{h} + \epsilon_{xx}(h,h) \\ \frac{u_y(h,0) - u_y(h,h)}{h} + \epsilon_{yy}(h,h) \\ \frac{u_x(h,0) - u_x(h,h) + u_y(0,h) - u_y(h,h)}{h} + 2\,\epsilon_{xy}(h,h) \\ \frac{u_x(0,h) - u_x(h,h)}{h} + \epsilon_{xx}(0,h) \\ \frac{u_y(0,0) - u_y(0,h)}{h} + \epsilon_{yy}(0,h) \\ \frac{u_x(0,0) - u_x(0,h) + u_y(0,h) - u_y(h,h)}{h} + 2\,\epsilon_{xy}(0,h) \end{pmatrix}. \tag{18}$$

From the 12 equations only 4 are linear independent (the rank of the coefficient matrix C is 4) and we can build the Moore–Penrose pseudo-inverse of C [16,17] to compute the unknown coefficients $\boldsymbol{g}$:

$$\boldsymbol{g} = \mathsf{C}^{-1}\boldsymbol{r}. \tag{19}$$

The matrix C^{-1} is given by

$$\mathsf{C}^{-1} = \begin{pmatrix} \frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & \frac{h}{16} & 0 & 0 \\ 0 & 0 & \frac{h}{16} & 0 & 0 & \frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & -\frac{h}{16} \\ 0 & \frac{h}{16} & 0 & 0 & \frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 \\ 0 & 0 & \frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & -\frac{h}{16} & 0 & 0 & \frac{h}{16} \end{pmatrix}, \tag{20}$$

and the second set of coefficients is given with

$$\begin{aligned} c_x &= \frac{h}{16}\left[\epsilon_{xx}(0,0) + \epsilon_{xx}(0,h) - \epsilon_{xx}(h,0) - \epsilon_{xx}(h,h)\right] \\ d_y &= \frac{h}{16}\left[\epsilon_{yy}(0,0) + \epsilon_{yy}(h,0) - \epsilon_{yy}(0,h) - \epsilon_{yy}(h,h)\right] \\ c_y &= \frac{h}{8}\left[\epsilon_{xy}(0,0) + \epsilon_{xy}(h,0) - \epsilon_{xy}(0,h) - \epsilon_{xy}(h,h)\right] + \frac{1}{8}\left[u_y(0,0) + u_y(h,h) - u_y(0,h) - u_y(h,0)\right] \\ d_x &= \frac{h}{8}\left[\epsilon_{xy}(0,0) + \epsilon_{xy}(0,h) - \epsilon_{xy}(h,0) - \epsilon_{xy}(h,h)\right] + \frac{1}{8}\left[u_x(0,0) + u_x(h,h) - u_x(0,h) - u_x(h,0)\right]. \end{aligned} \tag{21}$$

Eq. (21) shows the added value by using the information contained in the second order moments. If one would not use this information, but compute the elements of the strain rate tensors by finite differences using the four nodes of the cell, the coefficients c_x, c_y, d_y and d_x would be zero.

4. Check for consistency: Arbitrary flow field

To check whether the procedure proposed for the reconstruction of the velocity field yields second order accurate results, we consider an arbitrary, smooth flow field. The Taylor expansion of the flow field $\mathbf{u} = (f_x(x,y), f_y(x,y))$ around the point

(x_0, y_0) is given by

$$\begin{aligned} u_\alpha &= u_\alpha(x_0, y_0) + \partial_x u_\alpha(x_0, y_0)\,(x - x_0) + \partial_y u_\alpha(x_0, y_0)\,(y - y_0) \\ &\quad + \frac{1}{2}\partial_{x,x} u_\alpha(x_0, y_0)\,(x - x_0)^2 + \frac{1}{2}\partial_{y,y} u_\alpha(x_0, y_0)\,(y - y_0)^2 \\ &\quad + \partial_{x,y} u_\alpha(x_0, y_0)\,(x - x_0)\,(y - y_0) + \cdots, \quad \alpha = x, y. \end{aligned} \tag{22}$$

We substitute the Taylor expanded flow field (22) in Eq. (16) to compute the coefficients $(\{a_i, i = 0, \ldots, 3\}, \{b_i, i = 0, \ldots, 3\})$ and in Eq. (21) to compute the coefficients $\boldsymbol{g} = (\{c_i, i = 0, \ldots, 1\}, \{d_i, i = 0, \ldots, 1\})$. The difference between the interpolated flow field given by Eq. (13) and the Taylor expanded flow field (22) is then

$$u_{a,\alpha}(x, y) - u_\alpha(x, y) = \mathcal{O}\left(h^3\right), \quad \alpha = x, y. \tag{23}$$

This shows that the interpolation (13) together with the coefficients (16) and (21) is exact up to second order.

5. Conclusions and outlook

We have proposed a new method for the reconstruction of the flow field in a cell composed of four lattice nodes for the d2q9 model. This method can be the base of a more local grid refinement algorithm for the LB method or it can be used to implement higher order boundary conditions. The extension to three space dimensions is straightforward.

Acknowledgement

The authors would like to thank Dr. Martin Geier for fruitful discussions.

Appendix. Transformation matrix M

The Transformation matrix M is given with

$$\mathsf{M} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & c & 0 & -c & 0 & c & -c & -c & c \\ 0 & 0 & c & 0 & -c & c & c & -c & -c \\ -2\,c^2 & c^2 & c^2 & c^2 & c^2 & 4\,c^2 & 4\,c^2 & 4\,c^2 & 4\,c^2 \\ 0 & c^2 & -c^2 & c^2 & -c^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & c^2 & -c^2 & c^2 & -c^2 \\ 0 & -c^3 & 0 & c^3 & 0 & 2\,c^3 & -2\,c^3 & -2\,c^3 & 2\,c^3 \\ 0 & 0 & -c^3 & 0 & c^3 & 2\,c^3 & 2\,c^3 & -2\,c^3 & -2\,c^3 \\ c^4 & -2\,c^4 & -2\,c^4 & -2\,c^4 & -2\,c^4 & 4\,c^4 & 4\,c^4 & 4\,c^4 & 4\,c^4 \end{bmatrix}. \tag{A.1}$$

References

[1] X. He, Error analysis for the interpolation-supplemented lattice-Boltzmann equation scheme, Internat. J. Modern Phys. C 8 (4) (1997) 737–745.

[2] O. Filippova, D. Hänel, Boundary-fitting and local grid refinement for LBGK models, Internat. J. Modern Phys. C 8 (1998) 1271.

[3] B. Crouse, E. Rank, M. Krafczyk, J. Tölke, A LB-based approach for adaptive flow simulations, Internat. J. Modern Phys. B 17 (2002) 109–112.

[4] D. Yu, R. Mei, W. Shyy, A multi-block lattice Boltzmann method for viscous fluid flows, Internat. J. Numer. Methods Fluids 39 (2) (2002) 99–120.

[5] U. Frisch, D. d'Humières, B. Hasslacher, P. Lallemand, Y. Pomeau, J.-P. Rivet, Lattice gas hydrodynamics in two and three dimensions, Complex Systems 1 (1987) 75–136.

[6] S. Chen, G.D. Doolen, Lattice Boltzmann method for fluid flows, Annu. Rev. Fluid Mech. 30 (1998) 329–364.

[7] Y.H. Qian, D. d'Humières, P. Lallemand, Lattice BGK models for Navier–Stokes equation, Europhys. Lett. 17 (1992) 479–484.

[8] D. d'Humières, Generalized lattice-Boltzmann equations, in: B.D. Shizgal, D.P. Weave (Eds.), Rarefied Gas Dynamics: Theory and Simulations, in: Prog. Astronaut. Aeronaut., vol. 159, AIAA, Washington, DC, 1992, pp. 450–458.

[9] P. Lallemand, L.-S. Luo, Theory of the lattice Boltzmann method: Dispersion, dissipation, isotropy, Galilean invariance, and stability, Phys. Rev. E 61 (6) (2000) 6546–6562.

[10] D. d'Humières, I. Ginzbург, M. Krafczyk, P. Lallemand, L.-S. Luo, Multiple-relaxation-time lattice Boltzmann models in three-dimensions, Phil. Trans. R. Soc. Lond. A 360 (2002) 437–451.

[11] J. Tölke, A thermal model based on the lattice Boltzmann method for low Mach number compressible flows, J. Comput. Theoret. Nanosci. 3 (4) (2006) 579–587.

[12] S. Chapman, T.G. Cowling, The Mathematical Theory of Non-Uniform Gases, Cambridge University Press, 1990.

[13] T. Inamuro, M. Yoshino, F. Ogino, Accuarcy of the lattice Boltzmann method for small Knudsen number with finite Reynolds number, Phys. Fluids 9 (1997) 3535.

[14] M. Junk, A. Klar, L.-S. Luo, Asymptotic analysis of the lattice Boltzmann equation, J. Comput. Phys. 210 (2005) 676.

[15] I. Ginzburg, D. d'Humières, Multireflection boundary conditions for lattice Boltzmann models, Phys. Rev. E 68 (2003) 066614.

[16] E.H. Moore, On the reciprocal of the general algebraic matrix, Bull. Amer. Math. Soc. 26 (1920) 394–395.

[17] R. Penrose, A generalized inverse for matrices, Proc. Cambr. Philo. Soc. 51 (1955) 406–413.